Bamisaye and Ojo, Sensor Netw Data Commun
2016, 5:2
DOI: 10.4172/2090-4886.1000141

International Journal of
Sensor Networks and Data Communications

Research Article Open Access

A Mobile Phone Controlled Electronic Immobilizer: An Application to Automobile Engineering

Ayodeji J Bamisaye* **and Alaba J Ojo**

Department of Electrical and Electronic Engineering, The Federal Polytechnic, Ado-Ekiti, Nigeria

***Corresponding author:** Ayodeji J Bamisaye, Department of Electrical and Electronic Engineering The Federal Polytechnic, Ado-Ekiti, Nigeria, Tel: 2347033657792; E-mail: ayobamisaye@gmail.com

Received date: April 23, 2016; **Accepted date:** May 11, 2016; **Published date**: May 17, 2016

Copyright: ©2016 Bamisaye AJ, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

There is a need for security to be provided for the automobiles due to the alarming rate of theft action across the globe. A system that reduce or even eliminate the menace on automobiles need to be developed. The design of automobile immobilizer using mobile phone is efficient in transmitting signal through the mobile phone that has both F (fast) and M (meter) buses network configurations to various destinations of users where communication network is available across the globe. The work was achieved with the aid of a specially designed microcontroller (PIC18F2550) that communicates directly with the mobile phone and some components like Schmitt Inverting Gate (74HC14N), Regulator LM7805 and LM317, NPN Transistor BC337, 12 Volt DC Relays and Liquid crystal display panel (LCD). It imbibed the operation of Short Message Service (SMS) and Communication is being established through a 15characters code (xxxxxxxxxxxxxx) which is secretive and privately used by the owner to enables and disables such automobile.

Keywords: Microcontroller; Automobile; Security; Code; Short message service (SMS)

Introduction

The most important need of man after food is safety. This can be defined as the security of lives and properties. But critically examining the present world, it will be discovered that the vital need of man is seriously menaced. Base on the level of crime and the geometrical increment in crime rate, it has become paramount on human mind the question of how to put an end to the roaring nuisance [1,2].

In view to proffer an effective and dependable ways of checkmating this problem, Research was made in the area of inventing and designing various forms of protecting systems. This system ranges from human to domestic security to mechanical systems [3-6]. Recent invention is in the area of electronics protection system and Biometric approach which is in form of tracking, alarm system, finger print recognition and face recognition [7-10]. The system is expensive and not every automobile user will be able to afford using such system because of its cost of design, development and maintenance.

Due to the fact that the world has become a global village, virtually everyone has access to mobile phone as a means of communication. With this development contact can be made to any particular individual through this means. Though Radio-Frequency Identification (RFID) does not need to be in line of sight with the receiver; has ability to pinpoint location and can store a lot of information. Notwithstanding, it has some disadvantages when compared with mobile system/GSM: it's more expensive; there are regulations about RFID guidelines; there is a private concern towards RFID and it may easily be intercepted even if it is encrypted. However, the application of mobile phone cannot be limited to conversation only. There are other ideas and innovation that can be derived from it, this will enhance its capabilities and functionalities. Technology such as Infra-red, Bluetooth, among others, which has developed in recent years show that improvements are possible and practicable, these will ease the way of life and means of communication [4].

Mobile phone has some features to send and receive SMS as well as multimedia messages apart from supporting voice calls; such messages include graphics, pictures animations among others. Text messages is very popular among mobile phone users because instant messages can be transmitted and received, this allow an individual to share ideas, opinions and other relevant information [11]. Base on this fact, we have made used of this concept to design a system that acts as a platform to receive messages which are the commands sent to control different systems, devices and appliances connected to the platform. The application of our suggested system is enormous in these modern technological days. With this system, there won't be any need to be physically present in order to control the system of a certain location.

Technology has improved so much in the last few decades in so much that it has made life more efficient and comfortable. The comfort of being able to remotely control systems or devices from one particular location has become essential as it saves time and effort. Furthermore it is a good protective measure against theft. Consequently, there arises a need to design such system in a systematic manner which we have tried to implement in this work. The proposed system is an extended approach to automating a control system. With the development of this system one can gain control over certain scheme that required regular attention and physical presence. The proposed system can remotely control automobile system by sending a text message from the mobile phone to the precise system. The goal of this work is to develop a system that allows a user to automatically control an automobile system with the aid of a mobile phone. The proposed move towards designing this system is to implement a microcontroller-based control system that receives instructions and commands from a mobile phone, process these commands and communicate the status of the system back to the mobile phone.

This Paper is sectioned into seven. Section two describes the Design Methodology, Sections three and four present the hardware and software overview respectively, section five describes the assembly of the system, while result and activation was discussed in section six, conclusion drawn from the paper was presented in section seven.

Design Methodology

Figure 1 showed the block diagram of the system. It is an illustration of system implementation. The transmitting section is the first mobile station, text messages that contain instructions and commands are sent by the subscriber to the second mobile station located in a particular environment where the control system is situated.

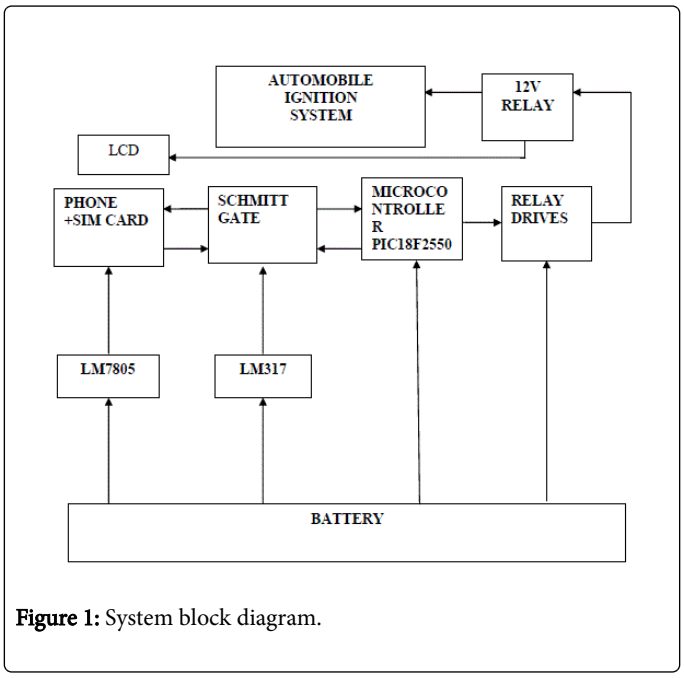

Figure 1: System block diagram.

The Subscriber Identification Module (SIM) stores the received messages. These are extracted by the microcontroller which will process accordingly in order to carry out the required operations.

The relay circuit is driven by the relay driver that is, buffer uln2003 which is responsible for switching the different appliances connected to the interface. The last message received is indicated on the Liquid Crystal Display (LCD). The operations performed by the micro-controller is also displayed, this in turn make the overall system user-friendly. To determine availability of utility supply, the scaled rectified mains supply is fed to micro-controller, therefore when the user try to call the mobile number integrated into the system, a message is sent by the system to indicat mains status automatically.

If the control unit is powered and operating properly, the process of controlling a system connected to the interface will progress through the following steps;

A text message is sent in form of a command to the receiver.

The sent message by the cell phone is received by the mobile phone receiver.

The sent message is decoded by the Mobile phone and sends the commands to the Microcontroller.

A command is issued by the Microcontroller to the automobile thereby switching on/off the selected devices.

The control unit/module will have the capacity to:

Automatically connect to the cellular network.

Receive text messages and be able to send and interpret the received messages and instructions to be delivered to the microcontroller.

Issue a command to the Automobile through the microcontroller.

Control the automobile.

Hardware Overview

The work comprised of eight basic units performing an interrelated function shown in Figure 1. It is being supplied by a 12v.dc source. The voltage regulator receives the impulse from the 12vdc battery and it is being regulated by the LM317, LM7805 voltage regulator. Moreover, the overall is connected to a relay drive. It is then soldered to the (PIC18F2550) Microcontroller which is the brain of the demobilizing system. The communication was set up between the mobile phone (RS232C interface) and the Microcontroller (PIC18F2550), for visual display, the entire circuit was connected to a Liquid Crystal Display.

A display means an optoelectronic device that can show a number, a hexadecimal digit, or any letter or number. The liquid crystal display (LCD) is a thin layer of "liquid crystal material" deposited between two plates of glass. An LCD is modelled as a capacitor, with connection to the common plane and the other side connected to the segment.

The considerations for this system will include a choice of networks, communication protocols and interfaces:

Communication protocol: the communication protocol used is SMS. The SMS is the most efficient because cellular communication and limited data are required to be sent and received.

O/I interface between microcontroller and devices: serial I/O is considered as options for connection between the Mobile phone receiver and the microcontroller. Using the microcontroller, a control circuit was implemented to control the electrical appliances.

Firmware Overview

Automobile immobilizer using mobile phone is a type of system being driven by soft code or password which is highly restricted to the owner alone for maximum protection and safety. This code is of 15characters that can ENABLE the automobile and at the same time DISABLE the engine from its normal operation. The system flowchart is shown in Figure 2. The system is connected in series across the automobile ignition and the car battery as shown in Figure 3.

F bus interface

A simple high level language tool in C was used as the tool for the software used. The sent message from the SIM location is extracted by the software at a regular interval and processes it in order to control the different appliances connected within the interface as shown in Figure 2. F-bus protocol was adopted to communicate with the mobile phone set [12]. We made use of a phone that has f-bus and m-bus connections that can be used to connect a phone to a PC or in this case a microcontroller. The connection can be used for controlling just about all functions of the phone, as well as uploading new firmware etc. this bus will allow SMS messages to be sent and received.

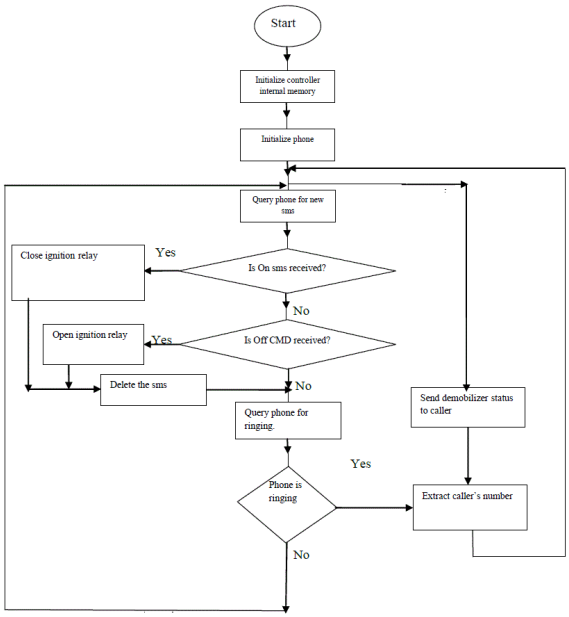

Figure 2: Program flow chart.

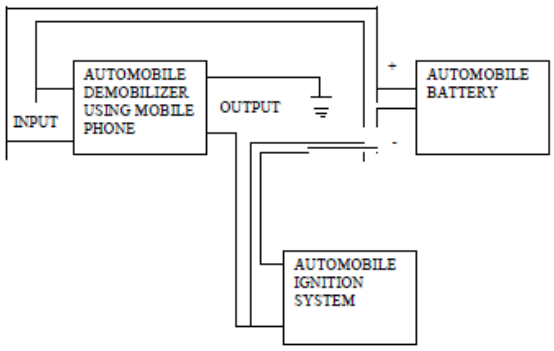

Figure 3: Automobile immobilizer connection diagram.

F-bus protocol and commands

We made use of f-bus protocol because its bi-directional serial type bus and it runs at 115,200bps, 8 data bits. The serial cable contains electronics for level conversion and therefore requires power. The serial cable is powered by setting the data terminal ready (DTR) pin and clearing the request to send (RTS) pin. The DTR pin has to be linked to a +3 v to 12 v supply and RTS to a -3 v to -12 v supply. This is achieved by using a max232 or related transceiver for the Rs232 transmitter and receiver pins and subsequently connecting the DTR pin on the serial cable to the v+ pin on the max232. The same process is carried out on the RTS but has to be connected to the v- pin on the max232. The v+ and v- pins are derived from internal charge pumps that double the input voltage, i.e., for a 5v max232, the v+ will produce +10v and the v- will be -10v.

Sample frame sent to the mobile phone (showed as a hex dump)

Byte: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15

Data: 1e 00 0c d1 00 07 00 01 00 03 00 01 60 00 72 d5

This sample frame is used to get the hardware and software version from a mobile phone. It is a good starting point to test if the implementation of the protocol is working.

Byte 0: all frames sent by cable will start with the character 0x1e first. This is the f-bus frame id. Cable is 0x1e and infrared is 0x1c.

Byte 1: this is the destination address. When sending data, it's the phone's device ID byte. In our case it's always 00 for the phone.

Byte 2: this is the source address. When sending data, it's the PC's device ID byte. In our case it's always 0x0c (terminal).

Byte 3: this is the message type or 'command'. 0xd1 is get HW and SW version.

Byte 4 and 5: byte 4 and 5 is the message length. In our case it is 7 bytes long. Byte 4 is the MSB and byte 5 is the LSB.

Byte 6: the data segment starts here and goes for 7 bytes in our case. As the phone is a 16 bit phone and therefore requires an even number of bytes. In this work, it is odd and thus the last byte will be a padding byte and the message will end at location 13.

The last byte in the data segment (byte 12 above) is the sequence number. The last 3 bits of these byte increases from 0 to 7 for each frame. This part needs to be sent back to the phone in the acknowledge frame. The checksum is calculated by XORing all the odd bytes and placing the result in the odd checksum location and then XORing the even bytes and then placing the result in the even byte. The phone receives and shows reply with the following data.

1e 0c 00 7f 00 02 d1 00 cf 71

1e 0c 00 d2 00 26 01 00 00 03 56 20 30

34 2e 34 35 0a 32 31 2d 30 36 2d 30 31

0a 4e 48 4d 2d 35 0a 28 63 29 20 4e 4d

50 2e 00 01 41 3f a4

The first line is an acknowledge command frame. The destination and source addresses are now swapped. This is because the mobile phone is now connecting. This message is two bytes long with the two bytes representing the message type received (0xd1) and the sequence number (0x00). The last two bytes are the checksum and should be checked to ensure the correctness of the data. After sending the two frames, an acknowledgement will be awaited by the phone. If the acknowledge frame is not sent the phone will repeat sending the data for three times before giving up. The second frame from the phone is the data requested. The message type is 0xd2. This is 'receive get HW and SW version'. this 38-byte (0x26) message should show 0x0003 "v" "firmware\n" "firmware date\n" "model\n" "(c) NMP." the last byte in the data is the sequence number. As with standard f-bus frames. The last two bytes in the frame are checksum bytes.

The received data without f-bus frame

01 00 00 03 56 20 30 34 2e 34 35 0a 32

31 2d 30 36 2d 30 31 0a 4e 48 4d 2d 35

0a 28 63 29 20 4e 4d 50 2e 00 01 41 00

03 v 0 4 . 4 5 \n 2 1 / 0 6 / 0 1 \n n h m - 5 \n (c) n m p .

Now, the acknowledge frame need to be sent back to the phone.

1e 00 0c 7f 00 02 d2 01 c0 7c 0x7f is the acknowledge frame's command. We only require sending a two-byte message so length is set to 0x02. The message contains the acknowledged message type (0xd2) and the sequence no. (0x01). the sequence number is made from the last 3 bits of the sequence number in the previous frame. The checksum needs to be calculated and sent.

Full SMS message frame: Sample frame sent to the mobile phone (showed as a hex dump) 98 bytes.

Byte: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35.

Data: 1e 00 0c 02 00 59 00 01 00 01 02 00 07 91 16 14 91 09 10 f0 00 00 00 00 15 00 00 00 33 0a 81 40 30 87 00 47.

SMS message centre - phone number - Byte:

36 37 38 39 40 41 42 43 44 45 46

47 48 49 50 51 52 53 54 55 56 57 58 59

60 61 62 63 64 65 66 67 68 69 70 71.

Data: 00 00 00 00 00 a7 00 00 00 00 00

00 c8 34 28 c8 66 bb 40 54 74 7a 0e 6a

97 e7 f3 f0 b9 0c ba 87 e7 a0 79 d9.

Start of message - hi all. This message was sent through f-bus.

Byte: 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97.

Data: 4d 07 d1 d1 f2 77 fd 8c 06 19 5b c2 fa dc 05 1a be df ec 50 08 01 43 00 7a 52.

F-bus frame header.

Byte 0: f-bus frame ID. We are on cable (0x1e).

Byte 1: destination address. Byte 2: source address.

Byte 3: message type or 'command'. 0x02 (SMS handling).

Byte 4 and 5: message length. In our case it is 0x0059 bytes long or 89 bytes in decimal.

(SMS) short message service frame header.

Byte 6 to 8: start of the SMS frame header. 0x00, 0x01, 0x00.

Byte 9 to 11: 0x01, 0x02, 0x00=send SMS message.

(SMSC) short message service centre (12 bytes).

Byte 12: SMS centre number length. 0x07 is 7 bytes long. This includes SMSC number.

Type and SMS centre phone number. Byte 13: SMSC number type, e.g. 0x81- unknown 0x91-international 0xa1- national.

1xxx iiii: where i is the numbering-plan- identification. 1ttt xxxx: where t is the type-of-number. byte 14 to 23: (octet format) SMS centre phone number for example +234803000000. (tpdu) transfer protocol data unit byte 24: message type xxxx xxx1=SMS submit - the short message is transmitted from the mobile station (MS) to the service centre (SC). xxxx xxx0=SMS deliver - the short message is transmitted from the SC to the MS. (refer to service provider 03.40 - 9.2.3 definition of the TPDU parameters). In our case it is 0x15=0001 0101 in binary. The message is SMS submit; reject duplicates, and validity indicator present.

Byte 25: message reference if SMS deliver and validity indicator used (not used in this case).

Byte 26: protocol id.

Byte 27: data coding scheme.

Byte 28: message size is 0x33 in hex or 51 bytes long in decimal. This is the size of the unpacked message.

Destination's phone number (12 bytes)

Byte 29: destination's number length.

Byte30: number type, e.g.0x81- unknown 0x91-international 0xa1- national

Byte 31 to 40: (octet format) destination's phone number Validity period (VP)

Byte 41: validity-period code. Time period during which the originator considers the short message to be valid.

Byte 42 to 47: service centre time stamp. For SMS-delivery. The SMS message (SMS-submit)

Byte 48 to 92: this is the SMS message packed into 7 bit characters. SMS point- to-point character packing.

Byte 93: always 0x00 The f-bus usual ending.

Byte 94: packet sequence number

Byte 95: padding byte - string is odd and requires padding byte to be even!

Byte 96 and 97: odd and even checksum bytes.

If the phone receives a valid frame it should reply with something like this below, to say it got the message.

Reply frame sent from the phone number (showed as a hex dump)

Byte: 00 01 02 03 04 05 06 07 08 09

Data: 1e 0c 00 7f 00 02 02 03 1c 72.

This is just like the above acknowledge command frame. The destination and source addresses are exchanged, because it's a frame from the phone to the microcontroller. This message is two bytes long with the first byte representing the message type received (0x02) while the next byte represent the sequence number (0x03). The last two bytes are the checksum and should be checked to make sure the data is correct. After a short time the phone will reply with a 'message sent' frame shown below.

Byte: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17.

Data: 1e 0c 00 02 00 09 01 08 00 02 64 12 00 01 44 00 3f 1e.

Byte 03: message type=0x02 - SMS handing.

Byte 04 and 05: message length=0x0009 - 9 bytes long

Byte 09: 0x02=message sent Byte 10 to 14: n/a

The microcontroller must then acknowledge the frame.

Byte: 00 01 02 03 04 05 06 07 08 09.

Data: 1e 00 0c 7f 00 02 02 04 10 79.

Receiving SMS message: The following frame should be sent from the microcontroller to Mobile phone.

Byte: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35.

Data: 1e 0c 00 02 00 59 01 08 00 10 02 10 00 07 91 16 14 91 09 10 f0 00 10 19 38 04 00 00 33 0b 91 16 04 73 08 70.

Byte: 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71.

Data: f4 70 40 32 25 30 30 82 22 74 45 4c 25 30 30 82 22 74 45 4c 74 7a 0e 6a 97 e7 f3 f0 b9 0c ba 87 e7 a0 79 d9.

Byte: 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97.

Data: 4d 07 d1 d1 f2 77 fd 8c 06 19 5b c2 fa dc 05 1a be df ec 50 08 01 45 00 4a 5c.

Byte 03: message type=0x02 - SMS handing.

Byte 04 and 05: message length=0x0059 - 89 bytes long.

Byte 09: 0x10=SMS message received.

Byte 10: 0x02=memory type=SIM.

Byte 11: 0x10=location where SMS message stored - required to delete SMS message. (TPDU) transfer protocol data unit.

Byte 24: 0x38.

Byte 25: 0x04.

Byte 26: protocol id.

Byte 27: data coding scheme.

Byte 28: message length. 0x33=51 bytes long.

The microcontroller must then acknowledge this frame like normal.

Byte: 00 01 02 03 04 05 06 07 08 09.

Data: 1e 00 0c 7f 00 02 02 05 10 78.

Assembling

After the appropriate preparation of the various sections of the block diagram (Figure 1) the components were mounted on the overboard. The various stages were constructed, assembled together as shown in Figure 4 and tests were conducted to determine the performances of each. The system shares the same battery with the automobile ignition which is 12dc voltage. Also the phone is constantly being supplied by the car battery. This can be achieved base on the parallel connection made between the system and the automobile. After the construction of the work was completed, the entire system section was package well in a rectangular casing made of waterproofed material (Figure 5). The casing is considerably small for easy installation.

Testing and Security Activation

The work was tested after construction had being made by connecting it to an automobile in parallel as shown in Figure 3.

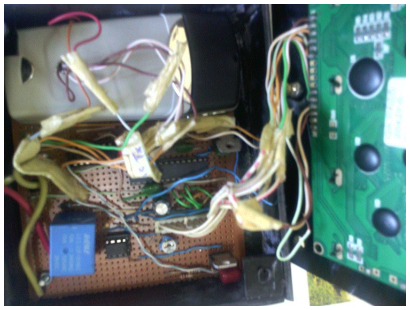

Figure 4: Assembling of components on Vero board.

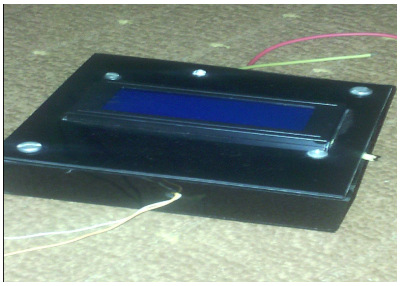

Figure 5: Picture of the mobile phone controlled immobilizer system.

The security code was then sent from the mobile phone as xxxxxxxxxxxxxx to demobilize the automobile i.e. disable the vehicle from moving. After a short while, another code was being sent from the same mobile phone as xxxxxxxxxxxxxx to enable the automobile to start working. The automobile Immobilizer using mobile phone sends SMS to the mobile phone that access and show update about the state of the automobile.

Conclusion

Automobile immobilizer using mobile phone is very important in control system engineering. It is highly efficient, reliable and effective because it is designed to offer maximum security to automobiles. Also the system may be improved to give maximum control of home appliances and full household electrical control. It is simple to operate and it maximises time with a constant voltage supply from the automobile. Nevertheless, the best way to reduce automobile theft across the globe is to install an automobile immobilizer using mobile phone in automobiles.

Based on the merit of the system; efficiency, accuracy, easy to install, user friendly and high reliability, and the consequent tests and certification, it is therefore recommended for use in the Automobile Engineering Systems.

References

1. Brown R (2004) The effectiveness of electronic immobilization: changing patterns of temporary and permanent vehicle theft. In: M.G. Maxeld and R.V. Clarke (eds.), Crime Prevention Studies 17, Criminal Justice Press, Monsey, NY.

2. Farrell G, Tseloni A, Tilley N (2011) The effectiveness of vehicle security devices and their role in the crime drop. Criminology and Criminal Justice 11: 21-35.
3. Kumar CN, Raghu Babu YV, Gamya A, Jainath P, Vijay M (2012) Design and Development of Activation and Controlling of Home Automation System VIA SMS through Microcontroller. International Journal of Engineering Research and Applications (IJERA) 2: 1349-1352.
4. Han HP, Tun HM (2014) Advanced Car Security System Using GSM. International Journal of Scientific and Research Publications 4: 1-5.
5. Ibrahim VM, Victor AA (2012) Microcontroller Based Anti-theft Security System Using GSM Networks with Text Message as Feedback. International Journal of Engineering Research and Development 2: 18-22.
6. Jayendra G, Kumarawadu S, Meegahapola L (2007) RFID-Based Antitheft Auto Security System with an Immobilizer. Second International Conference on Industrial and Information Systems, Sri Lanka.
7. Kiruthiga N, Latha LA (2014) Study of Biometric Approach for Vehicle Security System Using Fingerprint Recognition. International Journal of Advanced Research Trends in Engineering and Technology 1:10-16.
8. Patil1 SV, Sardeshmukh MM (2014) Face Recognition by Weber Law Descriptor for Anti-Theft Smart Car Security System. International Journal of Emerging Technology and Advanced Engineering 4: 224-228.
9. Powale PK, Zade GN (2014) Real time Car Antitheft System with Accident Detection using AVR Microcontroller; A Review. International Journal of Advance Research in Computer Science and Management Studies 2: 509-512 .
10. Montaser NR, Mohammad AA (2012) Intelligent AntiTheft and Tracking System for Automobiles. International Journal of Machine Learning and Computing 2: 88-92.
11. Sheikh IA, Sushil K (2011) Analysis and Performance of a Low Cost SMS Based Home Security System. International Journal of Smart Home 5: 15-24.
12. Wayne Peacock (2010) Nokia F-Bus Protocol.